# Análisis de datos: analizando mi distribución con tres alternativas de visualización

Guilherme Silveira
28/11/2020

Un análisis de datos inicial que hicimos de las notas de nuestros cursos en Alura llevó a la decisión de utilizar la estandarización de NPS para nuestros estudiantes. Usando NPS internamente, es posible analizar la historia de nuestros cursos.

Comenzando con los NPS más recientes de la base (usando pandas y una conexión con la base), mostramos los primeros N elementos de un grupo de cursos que seleccioné para analizar:

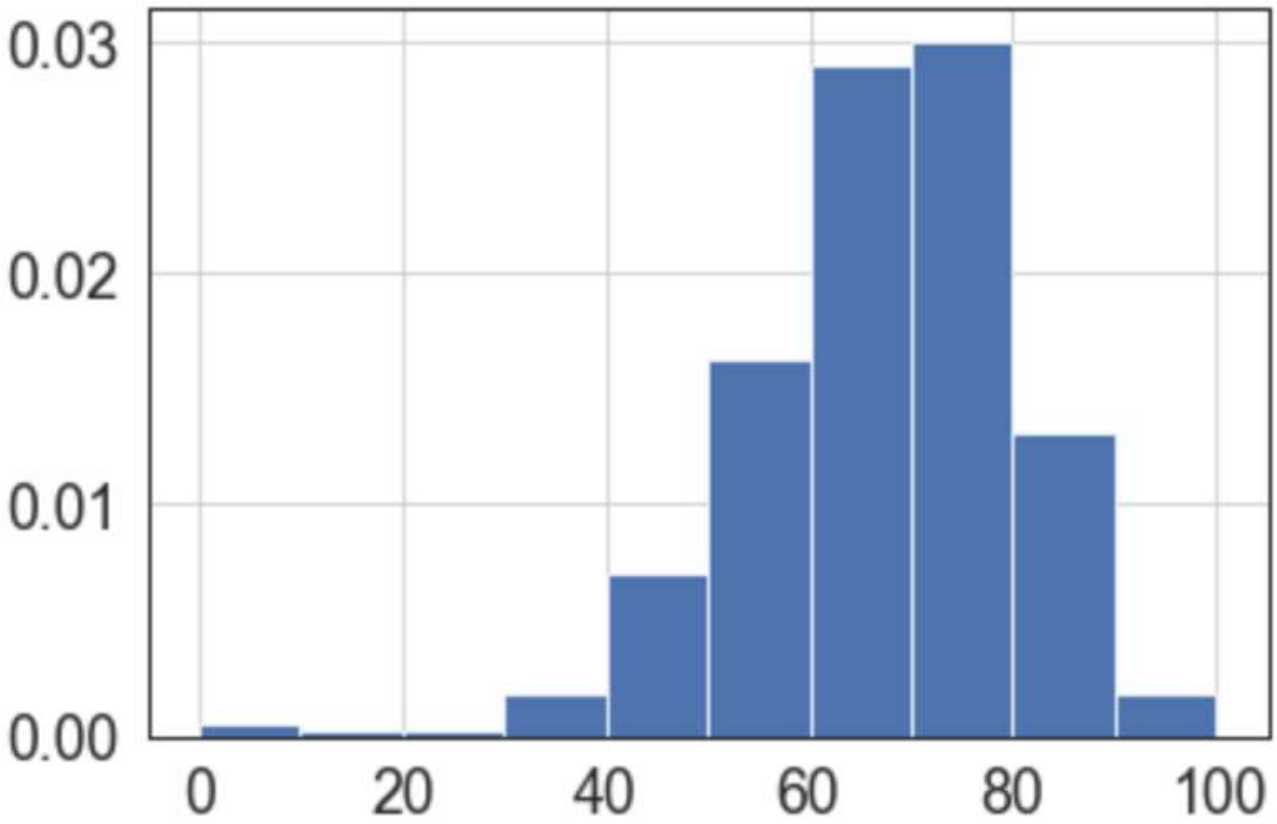
```
import pandas as pd
# ... abre la conexión ...

nps = pd.read_sql("select nps from __NPSRecent where /* periodo de tiempo por
nps.head()
```

| | nps |
|---|---|
| 0 | 66.6667 |
| 1 | 84.0285 |
| 2 | 72.6027 |
| 3 | 71.2329 |
| 4 | 59.4684 |

El NPS ya es un tipo de medida agregadora y es extraño calcular el promedio de NPS, así que probamos un gráfico simple, usando la función `plot` del mismo `pandas`. Solo pedimos que el histograma sea normalizado.

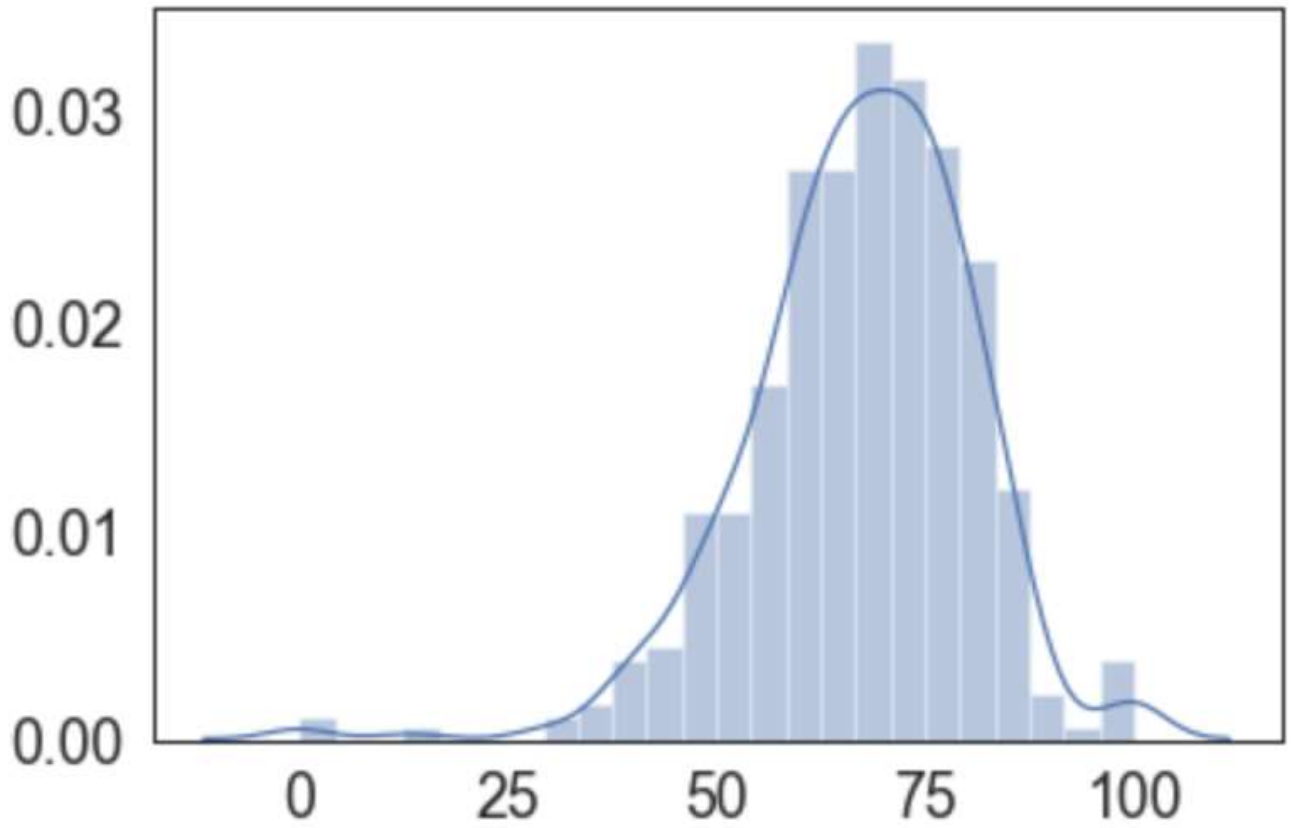
```
nps.hist (normed =True)
```



El gráfico es bonito y ya muestra que el propio NPS "mejoró" nuestra distribución ya que nuestra intención es provocar una diferencia más fuerte entre los mejores cursos y los demás. `pandas` utiliza el `matplotlib` detrás de escena y su configuración es de bajo nivel. Otra herramienta de gráficos conocida es `seaborn`. Importémosla y configurémosla para que use tamaño de fuente y tipo de colores diferentes:

```
import seaborn as sns
sns.set(color_codes=True)
sns.set_context("paper",font_scale=2)
sns.set_style("white")
```

Ahora, con base en los datos, pedimos que publique nuestra distribución:




Date cuenta cómo `seaborn` define un gráfico como de distribución, es decir, ya está pensando en la intención del gráfico y no en su tipo. Tanto es así que por defecto el gráfico de una distribución ya plota el histograma (barras) y la aproximación de una distribución (KDE).

Es curioso notar la asimetría de la distribución, notar como el lado derecho cae más rápidamente. Las colas también tienen pequeños “bumps”. ¿Nuestra distribución no se

comporta como una normal?

Pero estamos analizando un tipo de medida que se asemeja a un promedio, y la estadística clásica dice que tales promedios (en determinadas situaciones) se distribuyen como normal, ¿qué les está pasando a estas colas?

Vamos a pensar. Es básicamente imposible tomar solo notas 10 o 0 para siempre. Nada en el mundo suele ser perfecto, más aún, nada en el mundo suele ser perfecto para los ojos de todos los que miran, o estudian ese curso.

En algún momento de la historia, alguien dará una calificación que no sea 10. Incluso si es 9,99, no será 10. Por eso, a la larga es imposible mantener una puntuación de 10, como comenté en el post anterior. Tener solo una puntuación de 10, por lo tanto, indica una de dos cosas:

1. el curso aún no ha tenido suficientes alumnos, y aún no ha llegado alguien que todavía no ha sido tan fan del curso
2. hubo un previo filtrado, expulsando a los que no darían la máxima puntuación

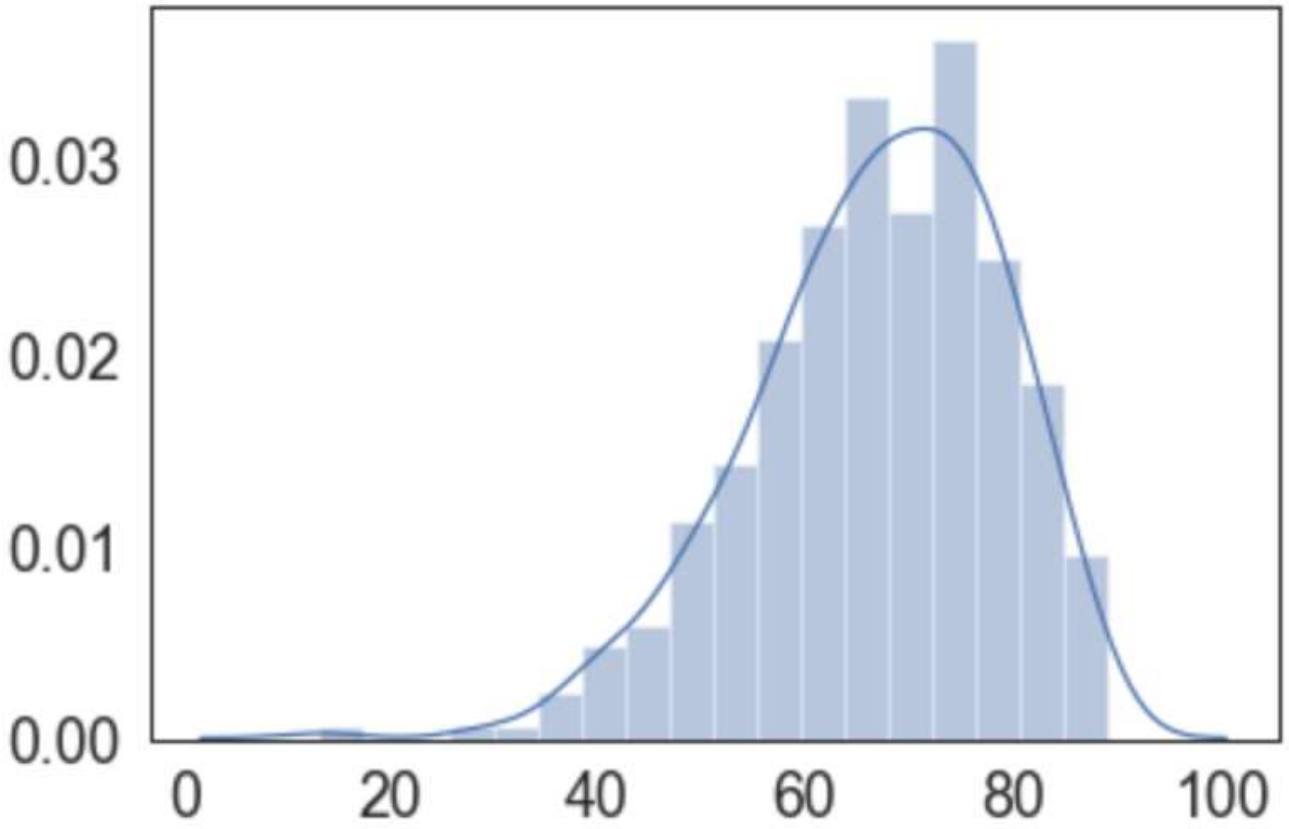
Entonces, los cursos cercanos a 100 son en realidad cursos con pocos votos, que en un momento u otro en el futuro tendrán al menos una calificación de 9. Lo mismo ocurre con un curso que tuvo solo dos evaluaciones, un primer grado 10 y luego un grado 6: la fórmula para estos dos grados da un NPS 0.

Un NPS 0 suele indicar un mal producto o, como en nuestro caso, simplemente que no tenía suficientes alumnos y fue una mala suerte que la persona que puntuaría 6 fuera el segundo alumno del curso. (es posible analizar la distribución de notas posibles y esperadas y entender qué tan común es el efecto de los dos extremos, y es super común, ya que el NPS esperado es alto).

Teniendo en cuenta estos puntos, es interesante considerar solo cursos con más de, por ejemplo, 50 notas, que ya se están escapando de estas anomalías:

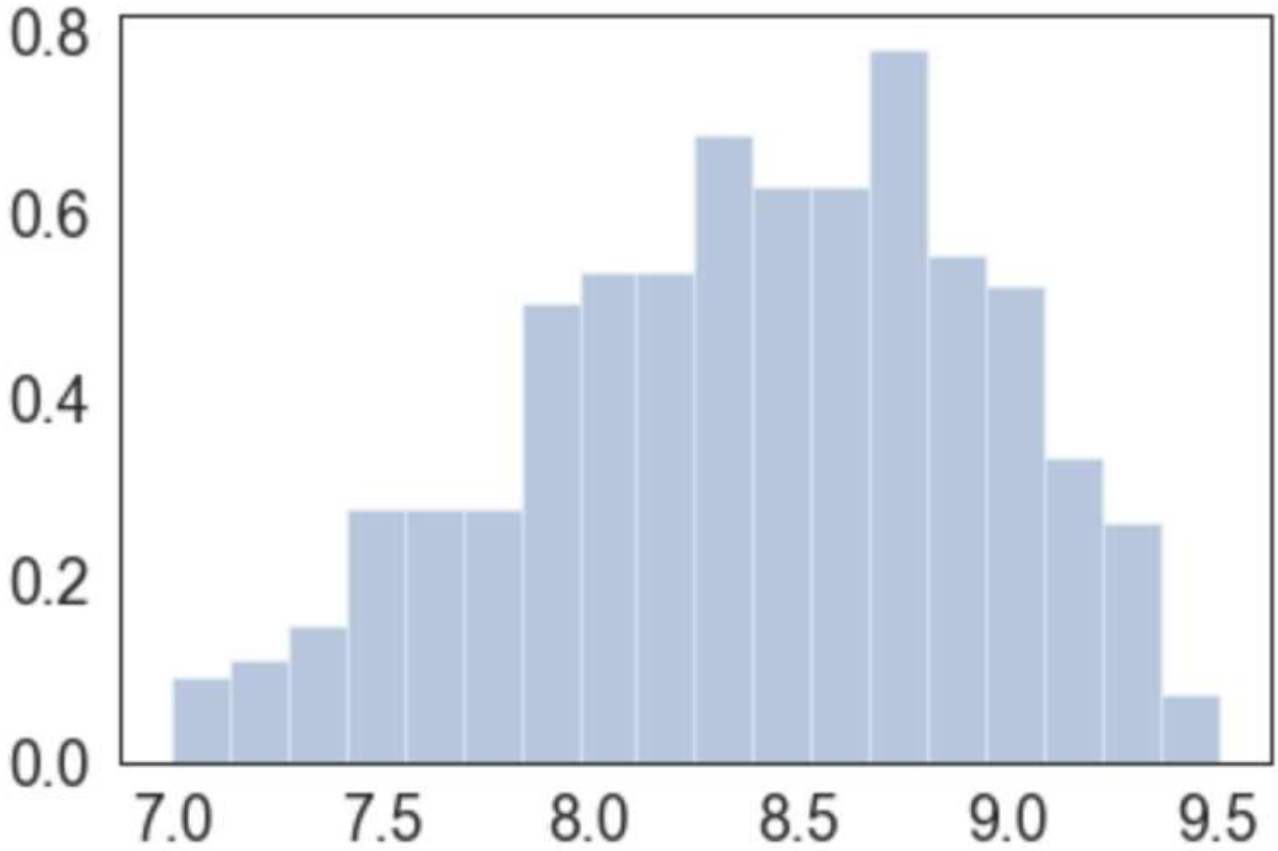
```
sql = """select nps from __NPSRecent where
                total_de_votos >= 50
                /* and periodo de tiempo por analizar */;"""
```

```
nps = pd.read_sql(sql, conexion)
```




Ahora lo que necesitamos es normalizar estos números NPS (que van de -100 a +100) y replotamos estos datos, ya enfocados en donde están básicamente todas las notas normalizadas:

```
sns.distplot(su_normalizacion(nps),
                  hist = True, norm_hist = True,
                  kde=False, hist_kws={"range": [7,9.5]})
```




Tenga en cuenta que con esta función de normalización que elegí tenemos una distribución que acerca los números a lo que espera un alumno. Es más visible que un curso normalizado de 9.5 nps se evaluó mejor que uno de 7.

Antes, estas notas estarían muy cerca. Toda la manipulación matemática se hizo con el objetivo de distanciar los cursos unos de otros, y bajar nuestros promedios, por más felices que estemos con promedios reales altos :)

¿El siguiente paso? Utilizar otra biblioteca para plotar nuestro histograma. Ahora probaremos el `altair`. Creamos un gráfico (chart) sobre nuestros datos `nps` y queremos dibujar barras (`mark_bar`).

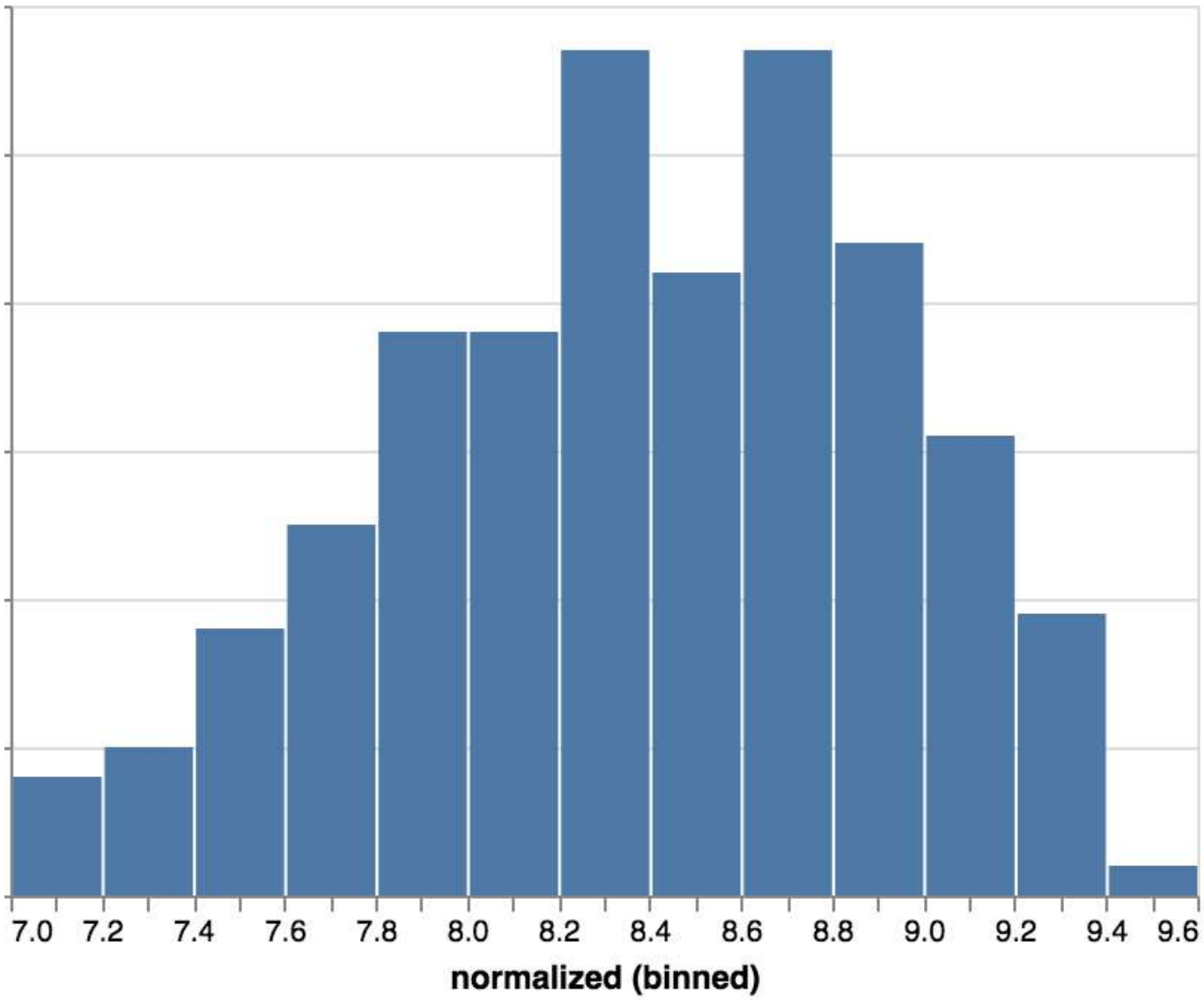
```
import altair as alt
# alt.renderers.enable('notebook') # si usa el jupyter notebook
alt.Chart(nps).mark_bar().encode(
   # ...
)
```

¿Qué pondremos dentro del gráfico? En el eje x vendrá el campo `nps` que es una variable cuantitativa (`Q`), y lo agruparemos en `20 bins`, como en un histograma. En el eje y, usaremos el recuento de elementos en cada `bin`:

```
import altair as alt
# alt.renderers.enable('notebook') # si usa el jupyter notebook
alt.Chart(nps).mark_bar().encode(
    alt.X('normalized:Q', bin=alt.BinParams(maxbins=20)),
    alt.Y('count()'),
)
```

Por último, digamos que nos interesa el rango de 7 a 9,6, recortando (`clamp`) el resto:

```
alt.Chart(nps).mark_bar().encode(
    alt.X('normalized:Q',
          bin=alt.BinParams(maxbins=20),
          scale=alt.Scale(domain=[7, 9.6], clamp=True)),
    alt.Y('count()'),
)
```




Tenga en cuenta que cada biblioteca ayudó de manera diferente. Mientras que `matplotlib` es la biblioteca de bajo nivel detrás de estos gráficos, utilicé los gráficos directamente desde `pandas` para plotar rápidamente visualizaciones y esbozos de lo que quiero ver.

A veces uso `seaborn` para vistas intencionales. Por supuesto, configurar la trama puede generar visualizaciones que ya resuelvan nuestro objetivo. Pero a veces la legibilidad del `altair` (declarativo) ayuda a mantener el código a largo plazo.

Con este abordaje y línea de pensamiento llegamos a la fórmula de normalización citada que usamos para distribuir "mejor" (mejor = notas más dispersas) las notas de nuestros cursos.

¿Qué tal aprender más sobre **Python** y análisis exploratorio de datos? Entonces, ¡Mira nuestros cursos de **Python para Data Science** aquí en Alura!

ARTÍCULOS DE TECNOLOGÍA > DATA SCIENCE

**En Alura encontrarás variados cursos sobre Data Science. ¡Comienza ahora!**

**SEMESTRAL**

**US$49,90**

un solo pago de US$49,90

- 218 cursos
- Videos y actividades 100% en Español
- Certificado de participación
- Estudia las 24 horas, los 7 días de la semana
- Foro y comunidad exclusiva para resolver tus dudas
- Acceso a todo el contenido de la plataforma por 6 meses

¡QUIERO EMPEZAR A ESTUDIAR!

Paga en moneda local en los siguientes países

ANUAL

US$79,90

un solo pago de US$79,90

- 218 cursos
- Videos y actividades 100% en Español
- Certificado de participación
- Estudia las 24 horas, los 7 días de la semana
- Foro y comunidad exclusiva para resolver tus dudas
- Acceso a todo el contenido de la plataforma por 12 meses

¡QUIERO EMPEZAR A ESTUDIAR!

Paga en moneda local en los siguientes países

Acceso a todos
los cursos

Estudia las 24 horas,
dónde y cuándo quieras

Nuevos cursos
cada semana

NAVEGACIÓN

PLANES
INSTRUCTORES
BLOG
POLÍTICA DE PRIVACIDAD
TÉRMINOS DE USO
SOBRE NOSOTROS
PREGUNTAS FRECUENTES

**¡CONTÁCTANOS!**

¡QUIERO ENTRAR EN CONTACTO!

**BLOG**

PROGRAMACIÓN
FRONT END
DATA SCIENCE
INNOVACIÓN Y GESTIÓN
DEVOPS

AOVS Sistemas de Informática S.A
CNPJ 05.555.382/0001-33

**SÍGUENOS EN NUESTRAS REDES SOCIALES**

**ALIADOS**

Empresa participante do SCALE UP ENDEAVOR

En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.

Google for Startups Alumni 2021

Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

## CURSOS

**Cursos de Programación**
Lógica de Programación | Java

**Cursos de Front End**
HTML y CSS | JavaScript | React

**Cursos de Data Science**
Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

**Cursos de DevOps**
Docker | Linux

**Cursos de Innovación y Gestión**
Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento